Speech Recognition

Tobias Gurdan

Seminar Human-Robot Interaction

Who I am

- Computer Science (B.Sc.) student, 5th semester
- From Bavaria
- Interests in
  - Graph Theory
  - Linear Algebra
  - Computer Vision and Graphics
  - Robotics, Machine Learning and AI

Motivation

Motivation

Motivation

Motivation

Outline

Outline

1. Difficulties in Speech Recognition
2. Speech Recognition Pipeline
    1) Preprocessing
    2) Hidden Markov Models
    3) Putting it together
3. Speech Recognition Today

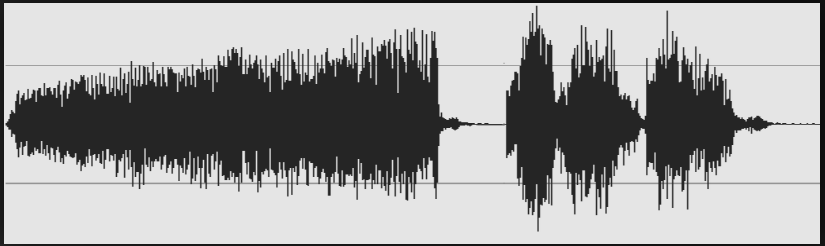
Difficulties in Speech Recognition

Difficulties in Speech Recognition

- **Optimal** vs. **corrupted** signal

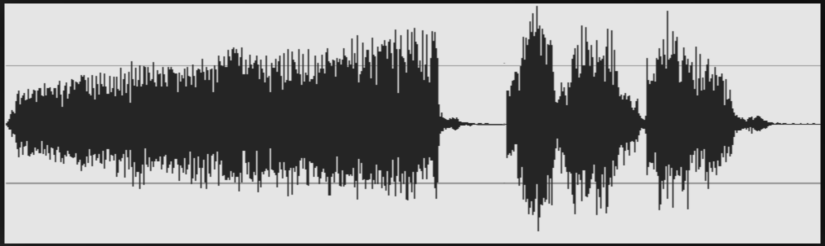
Difficulties in Speech Recognition

- **Optimal** vs. **corrupted** signal
  - Noise
  - Echo
  - Reverb
  - Sampling



Difficulties in Speech Recognition

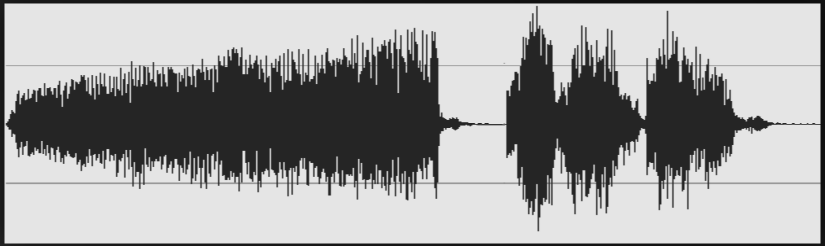
- **Optimal** vs. **corrupted** signal
  - Noise
  - Echo
  - Reverb
  - Sampling



- **Continuous** vs. **isolated** speech

Difficulties in Speech Recognition

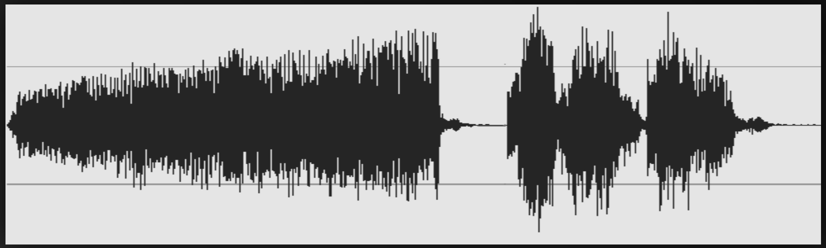
- **Optimal** vs. **corrupted** signal
  - Noise
  - Echo
  - Reverb
  - Sampling



- **Continuous** vs. **isolated** speech
- **Constrained** vs. **unconstrained** recognition

Difficulties in Speech Recognition

- **Optimal** vs. **corrupted** signal
  - Noise
  - Echo
  - Reverb
  - Sampling



- **Continuous** vs. **isolated** speech
- **Constrained** vs. **unconstrained** recognition
  - Natural language is very extensive
  - Almost perfect digit recognition (0-9)
  - Up to 45% error rates at 100.000 words (w/o relations)
  - Below 1% error rates for certain tasks (medical, law)

Difficulties in Speech Recognition

- Language **ambiguity**

Difficulties in Speech Recognition

- Language **ambiguity**

The tail of a dog

The tale of a dog

Difficulties in Speech Recognition

- Language **ambiguity**

The tail of a dog	The tale of a dog
It's easy to recognize speech	It's easy to wreck a nice beach

Difficulties in Speech Recognition

- Language **ambiguity**

The tail of a dog	The tale of a dog
It’s easy to recognize speech	It’s easy to wreck a nice beach
	It’s easy to wreck an ice peach

Difficulties in Speech Recognition

- Language **ambiguity**

The tail of a dog	The tale of a dog
It’s easy to recognize speech	It’s easy to wreck a nice beach
	It’s easy to wreck an ice peach
Oak wrap	...

Difficulties in Speech Recognition

- Language **ambiguity**

The tail of a dog	The tale of a dog
It’s easy to recognize speech	It’s easy to wreck a nice beach
	It’s easy to wreck an ice peach
Oak wrap	...
“e-set”	*b*, *c*, *d*, *e*, *g*, *p*, *t*, *v*, *z*

Difficulties in Speech Recognition

- Language **ambiguity**

The tail of a dog	The tale of a dog
It’s easy to recognize speech	It’s easy to wreck a nice beach
	It’s easy to wreck an ice peach
Oak wrap	...
“e-set”	*b*, *c*, *d*, *e*, *g*, *p*, *t*, *v*, *z*

- Speaker **variability**

Difficulties in Speech Recognition

- Language **ambiguity**

The tail of a dog	The tale of a dog
It's easy to recognize speech	It's easy to wreck a nice beach
	It's easy to wreck an ice peach
Oak wrap	...
"e-set"	*b*, *c*, *d*, *e*, *g*, *p*, *t*, *v*, *z*

- Speaker **variability**
  – Anatomy (sex, pitch)
  – Speed
  – Dialect

Speech Recognition Pipeline

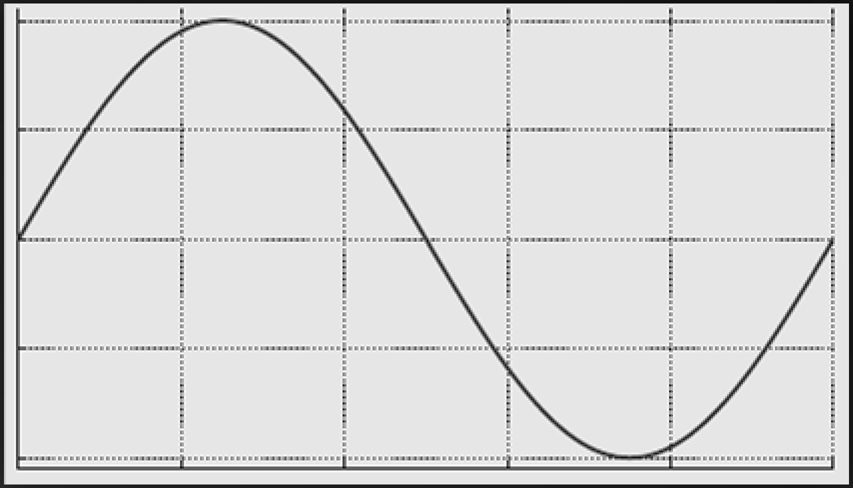
1) Preprocessing

Preprocessing

1. Recording and sampling

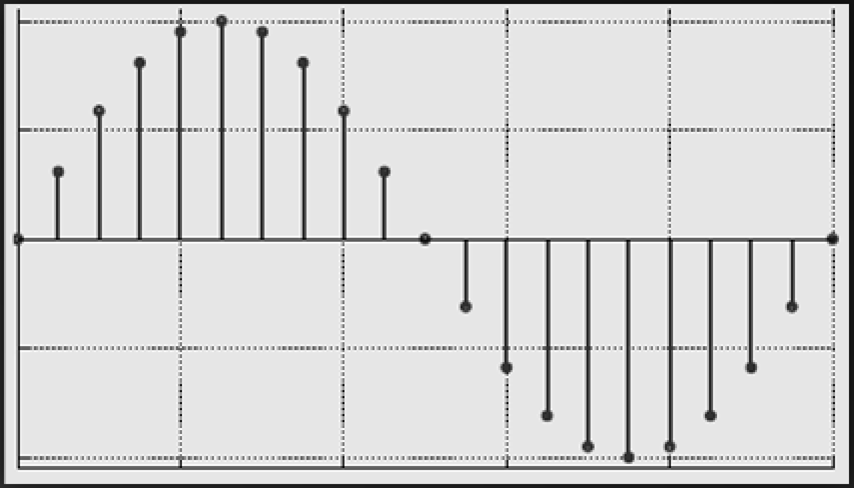
Preprocessing

1. Recording and sampling



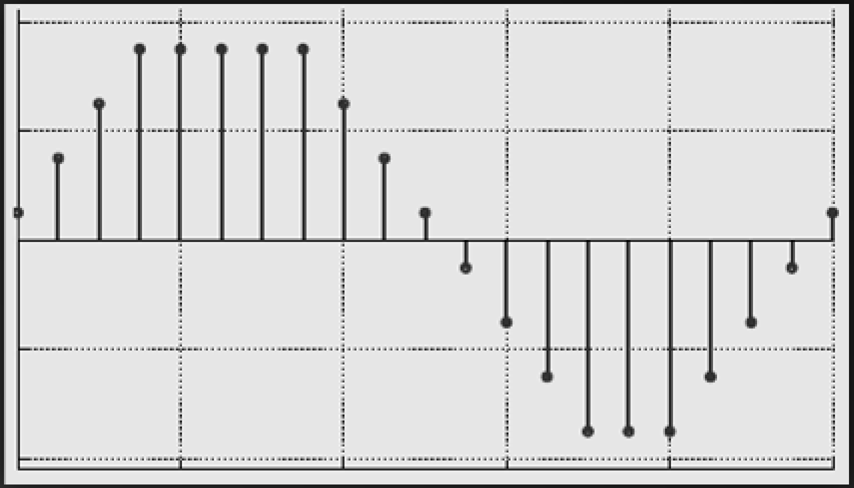
Preprocessing

1. Recording and sampling



Preprocessing

1. Recording and sampling



Preprocessing

1. Recording and sampling
2. Filtering

Preprocessing

1. Recording and sampling
2. Filtering
   - Speech enhancement

Preprocessing

1. Recording and sampling
2. Filtering
   - Speech enhancement
   - Cocktail party problem

Preprocessing

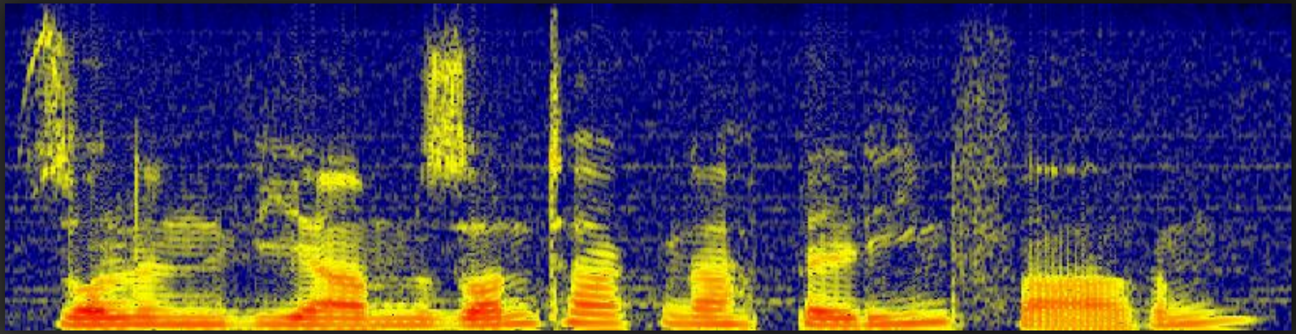
1. Recording and sampling
2. Filtering
3. Transformation

Preprocessing

1. Recording and sampling
2. Filtering
3. Transformation
    - Time domain (waveform)

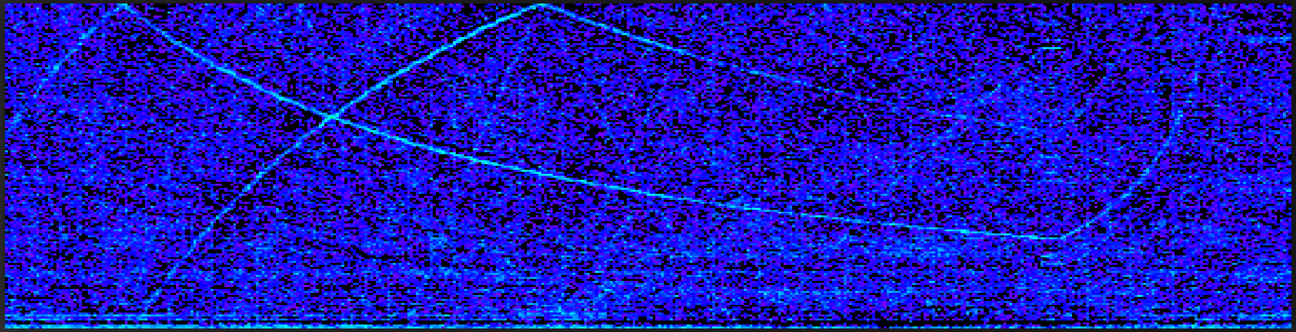
Preprocessing

1. Recording and sampling
2. Filtering
3. Transformation
   - Time domain (waveform)
   - Frequency domain (spectrum)



Preprocessing

1. Recording and sampling
2. Filtering
3. Transformation
   - Time domain (waveform)
   - Frequency domain (spectrum)
   - Quefrency domain (cepstrum)



Preprocessing

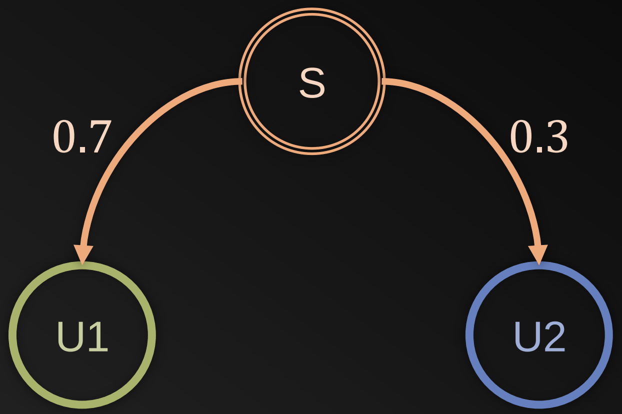
1. Recording and sampling
2. Filtering
3. Transformation
4. Feature vector

Speech Recognition Pipeline

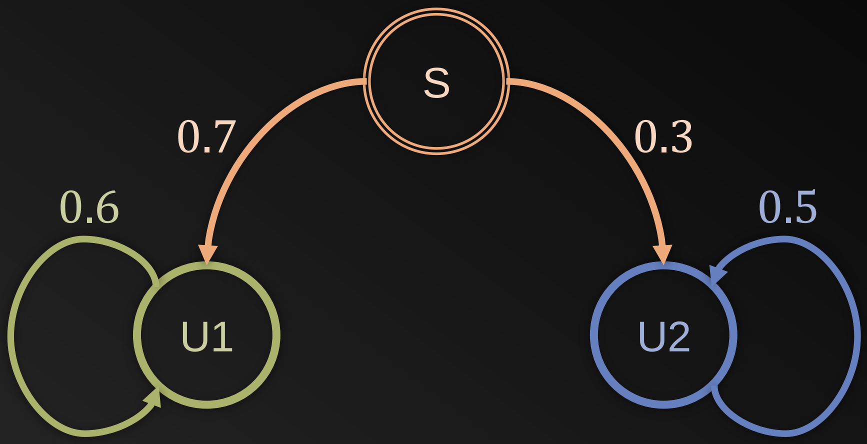
2) Hidden Markov Models

Hidden Markov Models

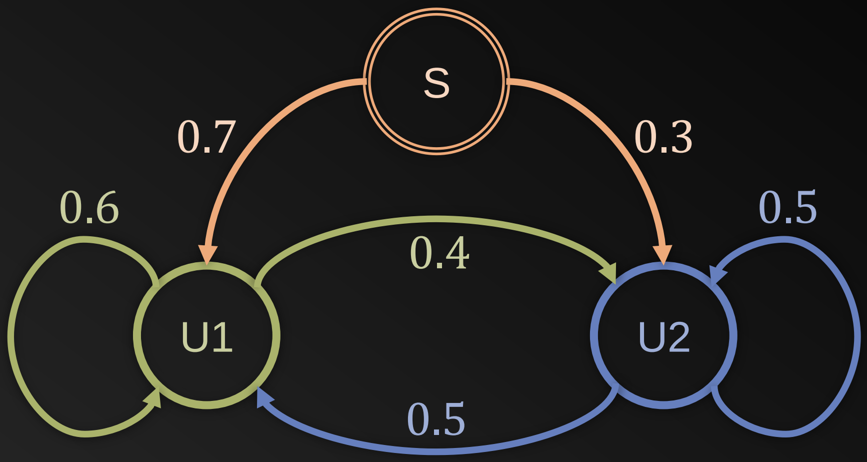
Hidden Markov Models



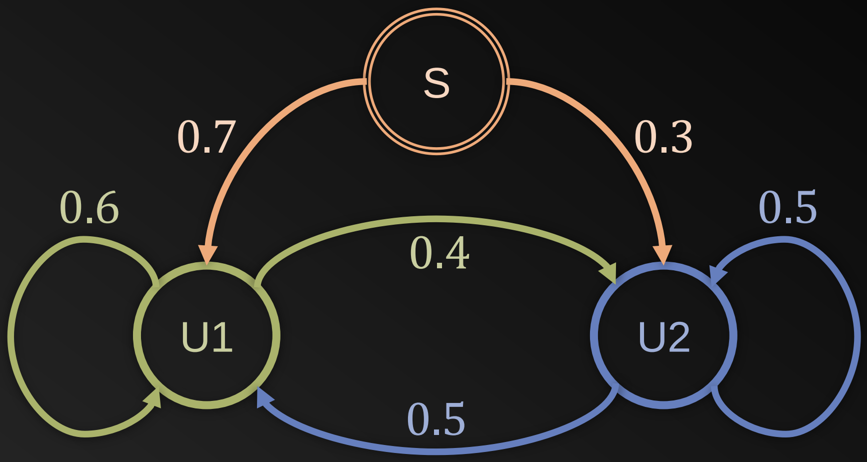
Hidden Markov Models



Hidden Markov Models



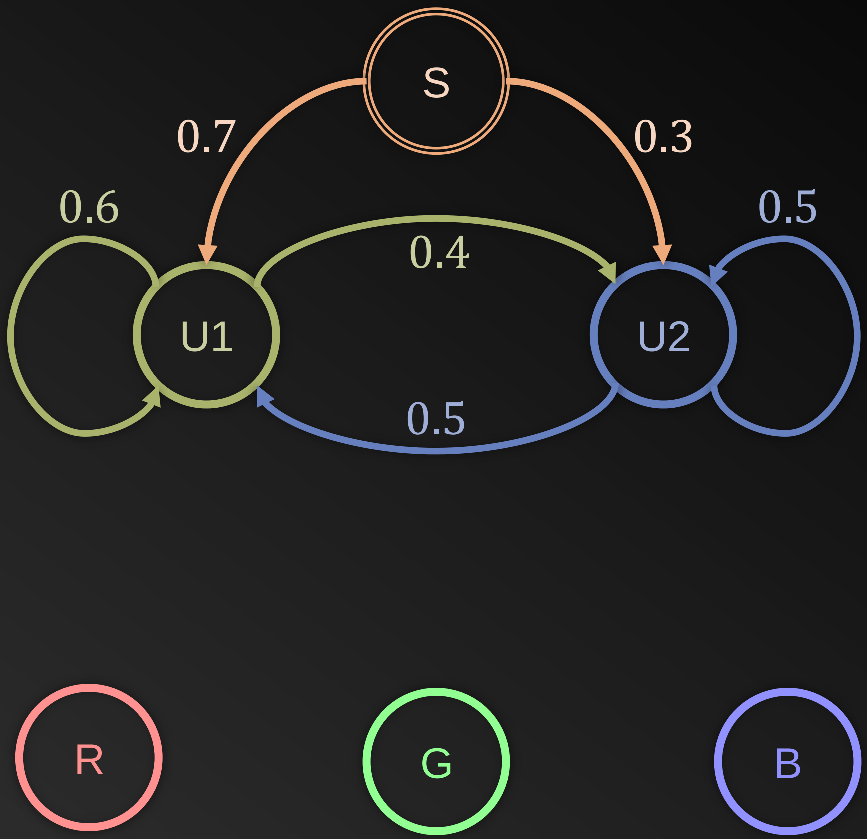
Hidden Markov Models



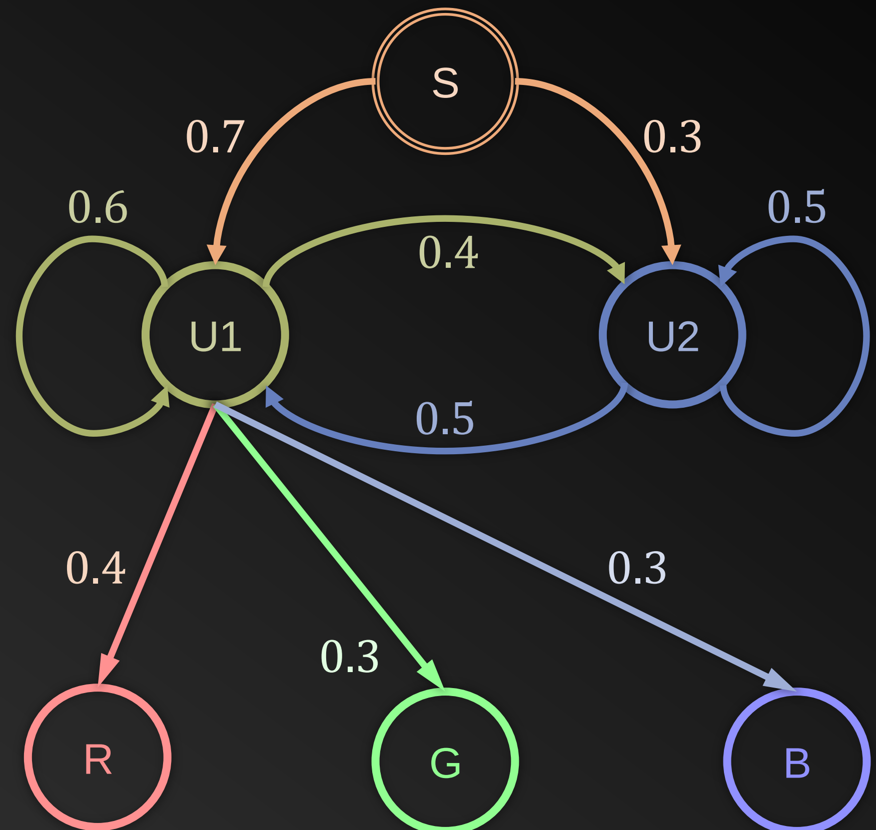
Markov property:

$$Pr(X_{i+1} = x | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = Pr(X_{i+1} = x | X_n = x_n)$$

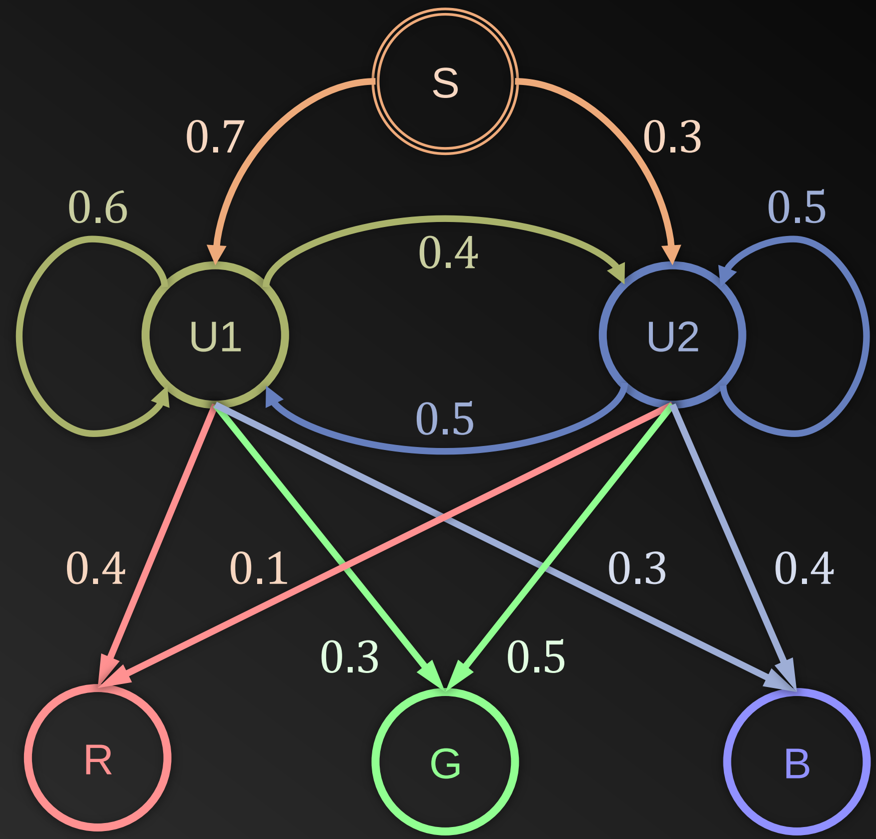
Hidden Markov Models



Hidden Markov Models

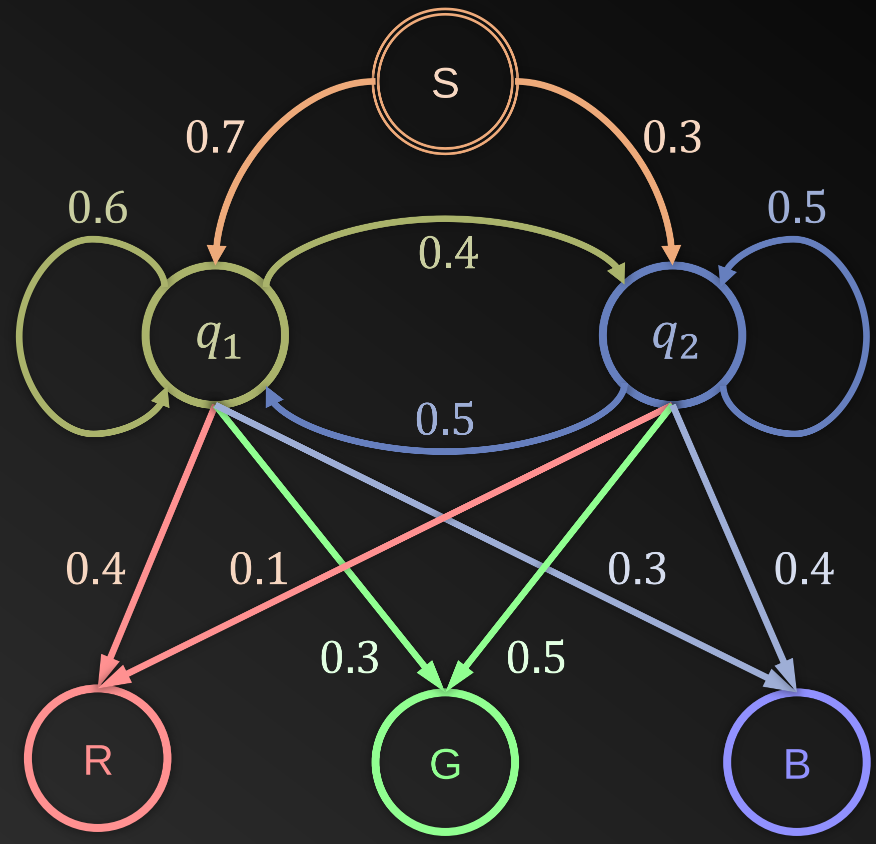


Hidden Markov Models



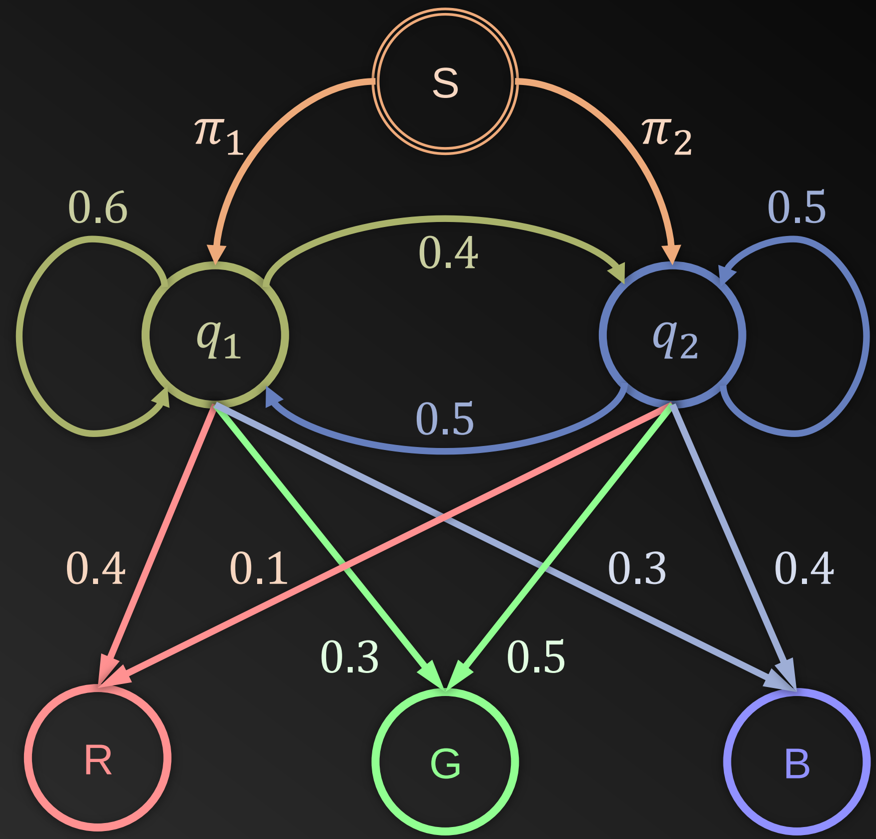
Hidden Markov Models

- $Q = \{q_1, \dots, q_N\}$



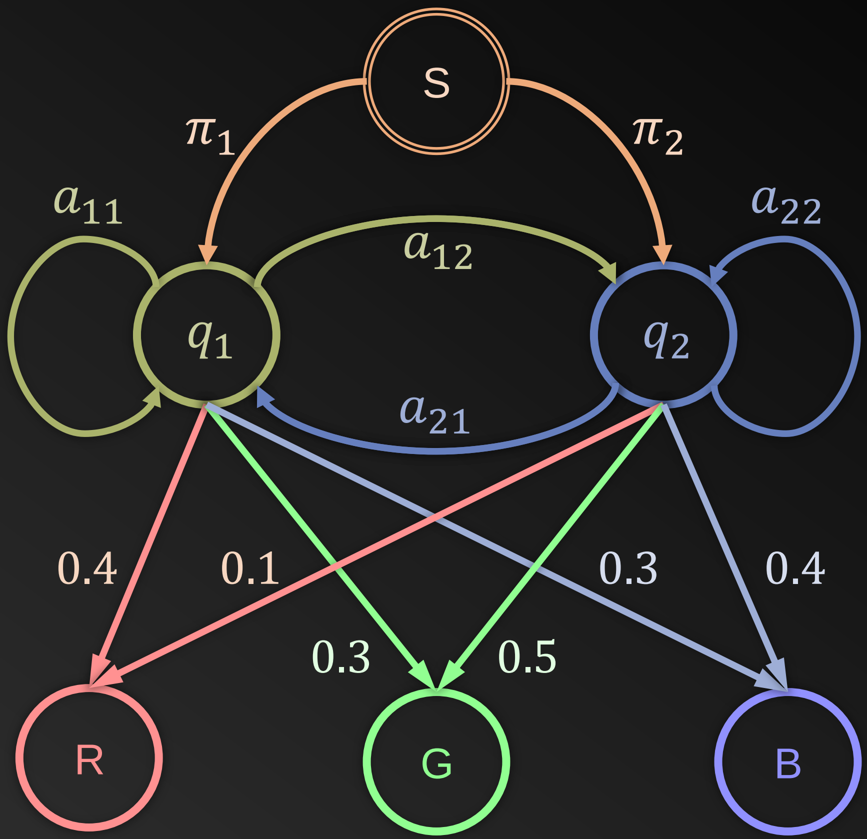
Hidden Markov Models

- $Q = \{q_1, \dots, q_N\}$
- $\Pi = (\pi_1 \quad \cdots \quad \pi_N)$



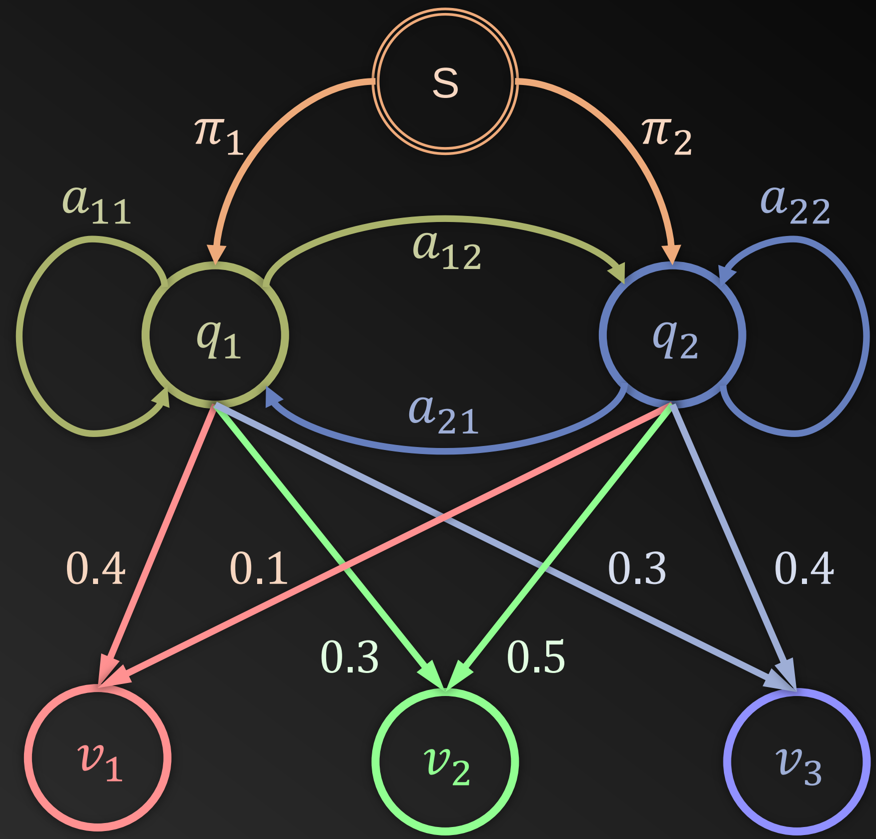
Hidden Markov Models

- $Q = \{q_1, \dots, q_N\}$
- $\Pi = \begin{pmatrix} \pi_1 & \cdots & \pi_N \end{pmatrix}$
- $A = \begin{pmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{N1} & \cdots & a_{NN} \end{pmatrix}$



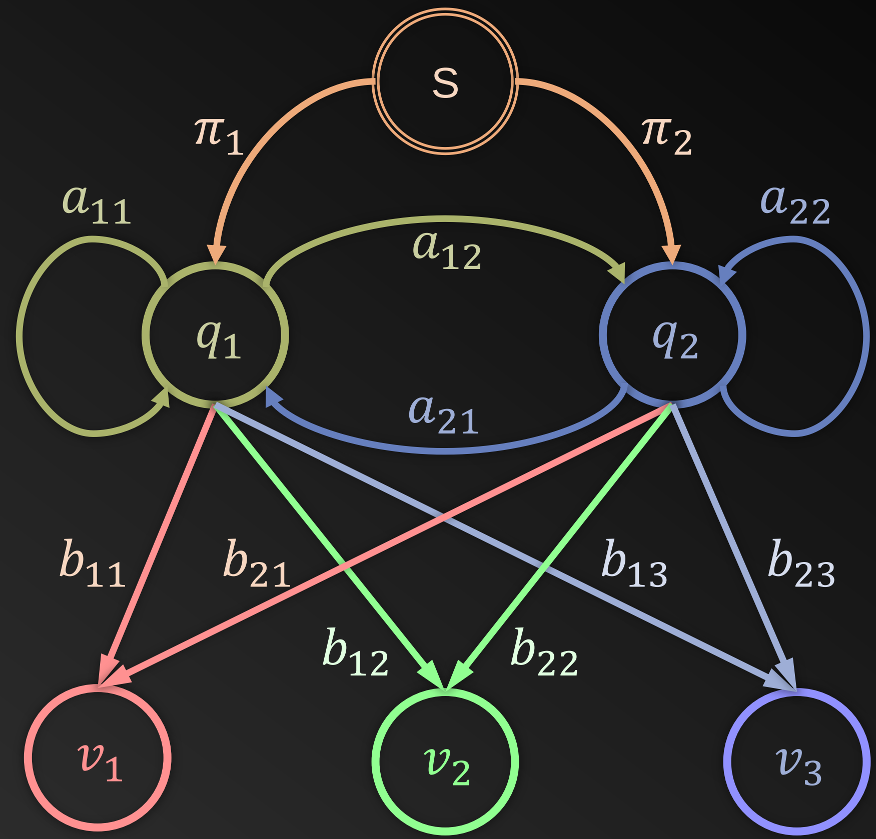
Hidden Markov Models

- $Q = \{q_1, \ldots, q_N\}$
- $\Pi = \begin{pmatrix} \pi_1 & \cdots & \pi_N \end{pmatrix}$
- $A = \begin{pmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{N1} & \cdots & a_{NN} \end{pmatrix}$
- $V = \{v_1, \ldots, v_M\}$



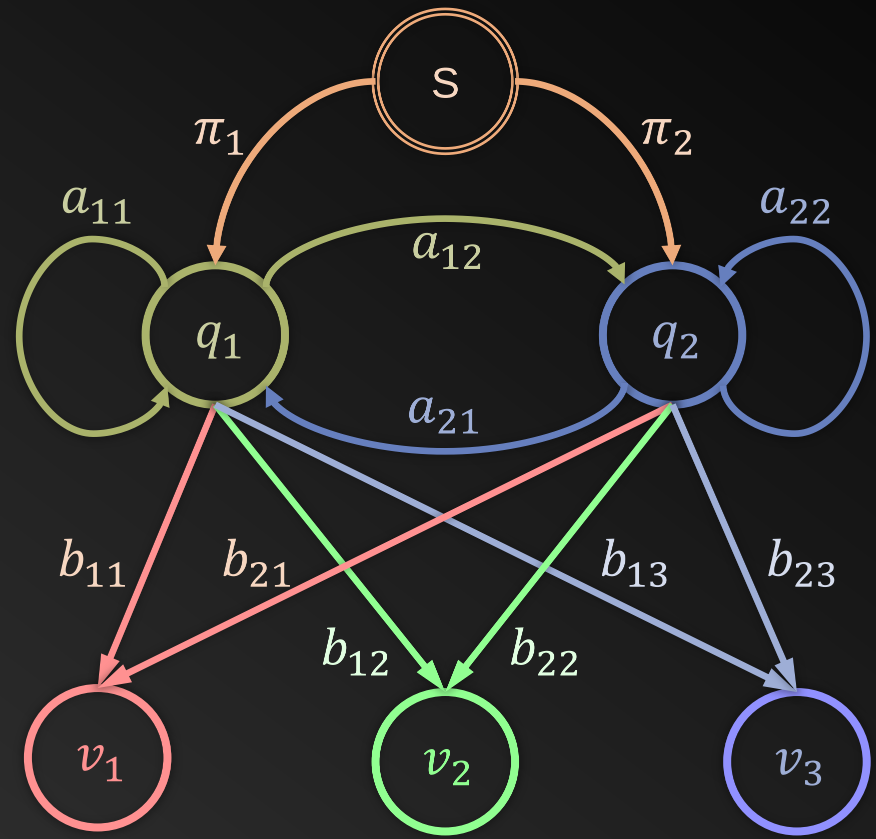
Hidden Markov Models

- $Q = \{q_1, \dots, q_N\}$
- $\Pi = (\pi_1 \quad \cdots \quad \pi_N)$
- $A = \begin{pmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{N1} & \cdots & a_{NN} \end{pmatrix}$
- $V = \{v_1, \dots, v_M\}$
- $B = \begin{pmatrix} b_{11} & \cdots & b_{1M} \\ \vdots & \ddots & \vdots \\ b_{N1} & \cdots & b_{NM} \end{pmatrix}$



Hidden Markov Models

- $Q = \{q_1, \dots, q_N\}$
- $\Pi = \begin{pmatrix} \pi_1 & \cdots & \pi_N \end{pmatrix}$
- $A = \begin{pmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{N1} & \cdots & a_{NN} \end{pmatrix}$
- $V = \{v_1, \dots, v_M\}$
- $B = \begin{pmatrix} b_{11} & \cdots & b_{1M} \\ \vdots & \ddots & \vdots \\ b_{N1} & \cdots & b_{NM} \end{pmatrix}$
- $\mu := (Q, V, \Pi, A, B)$



Hidden Markov Models

- The *hidden* part
  - State sequence is unknown
  - We only get to see the observation
  - $O = (o_1, \dots, o_T)$
  - e.g. $(red, green, red, blue)$

Hidden Markov Models

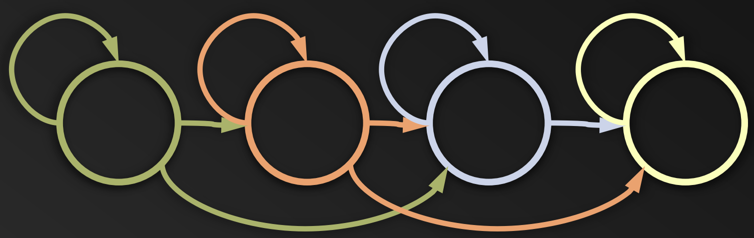
- Different HMM types

Hidden Markov Models

- Different HMM types
  - 3-state ergodic model

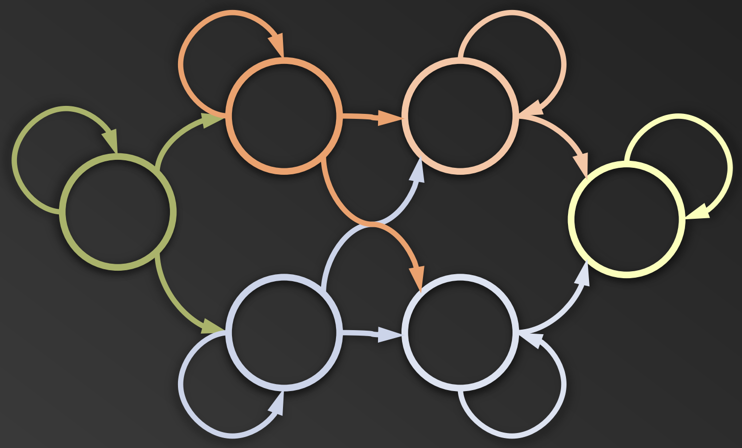
Hidden Markov Models

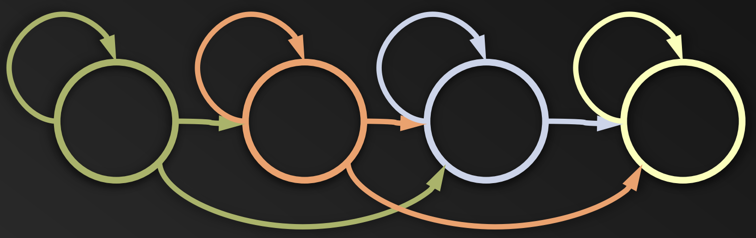
- Different HMM types
  - 3-state ergodic model
  - 4-state left-right model



Hidden Markov Models

- Different HMM types
  - 3-state ergodic model
  - 4-state left-right model
  - 6-state parallel path left-right model





Hidden Markov Models

- The three challenges

Hidden Markov Models

- The three challenges

(1) Given μ, calculate $Pr(O|\mu)$

Hidden Markov Models

- The three challenges

(1) Given μ, calculate $Pr(O|\mu)$

(2) Given μ and O, find $arg\ max_I\ Pr(O, I|\mu)$

Hidden Markov Models

- The three challenges

(1) Given μ, calculate $Pr(O|\mu)$

(2) Given μ and O, find $arg\ max_I\ Pr(O,I|\mu)$

(3) Given O, find best model μ

Hidden Markov Models

- The three challenges

(1) Given μ, calculate $Pr(O|\mu)$

  - ➢ Trellis graph brute-force search, $\mathrm{O}(T \cdot N^T)$

(2) Given μ and O, find $arg\ max_I\ Pr(O, I|\mu)$

(3) Given O, find best model μ

Hidden Markov Models

- The three challenges

(1) Given μ, calculate $Pr(O|\mu)$

- ➢ Trellis graph brute-force search, $\mathrm{O}(T \cdot N^T)$
- ➢ Forward-backward algorithm, $\mathrm{O}(T \cdot N^2)$

(2) Given μ and O, find $arg\ max_I\ Pr(O, I|\mu)$

(3) Given O, find best model μ

Hidden Markov Models

- The three challenges

(1) Given μ, calculate $Pr(O|\mu)$
  - ➢ Trellis graph brute-force search, $\mathrm{O}(T \cdot N^T)$
  - ➢ Forward-backward algorithm, $\mathrm{O}(T \cdot N^2)$

(2) Given μ and O, find $arg\ max_I\ Pr(O, I|\mu)$
  - ➢ Viterbi algorithm, $\mathrm{O}(T \cdot N^2)$

(3) Given O, find best model μ

Hidden Markov Models

- The three challenges

(1) Given μ, calculate $Pr(O|\mu)$

- ➢ Trellis graph brute-force search, $\mathrm{O}(T \cdot N^T)$
- ➢ Forward-backward algorithm, $\mathrm{O}(T \cdot N^2)$

(2) Given μ and O, find $arg\ max_I\ Pr(O, I|\mu)$

- ➢ Viterbi algorithm, $\mathrm{O}(T \cdot N^2)$

(3) Given O, find best model μ

- ➢ Baum-Welch algorithm, $\mathrm{O}(T \cdot N^3)$

Demo

Speech Recognition Pipeline

3) Putting it together

Putting it together

- Isolated word recognition

Putting it together

- Isolated word recognition
  - Vocabulary of size M

“one”

“two”

“three”

Putting it together

- Isolated word recognition
  - Vocabulary of size M
- Acquire data

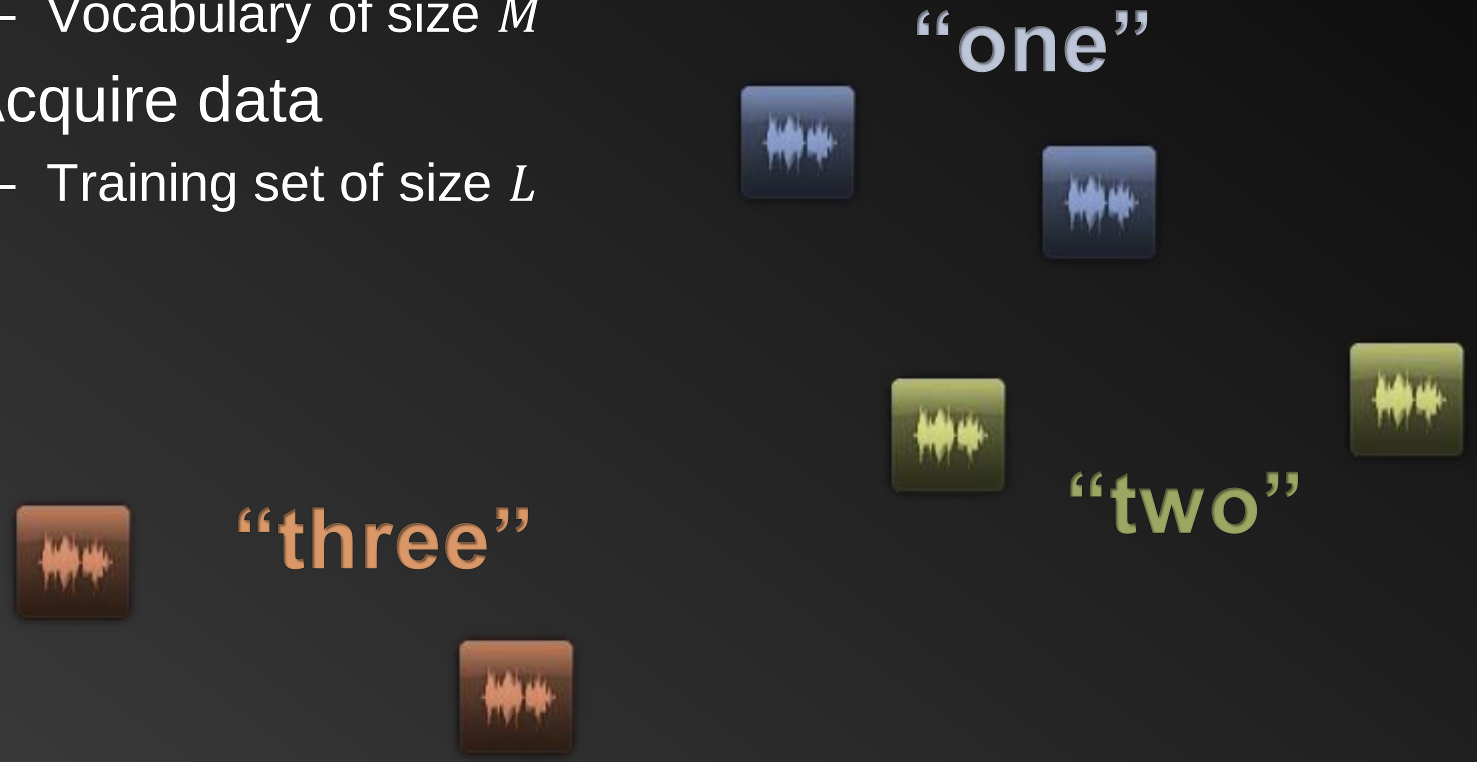
“one”

“two”

“three”

Putting it together

- Isolated word recognition
  - Vocabulary of size M
- Acquire data
  - Training set of size L

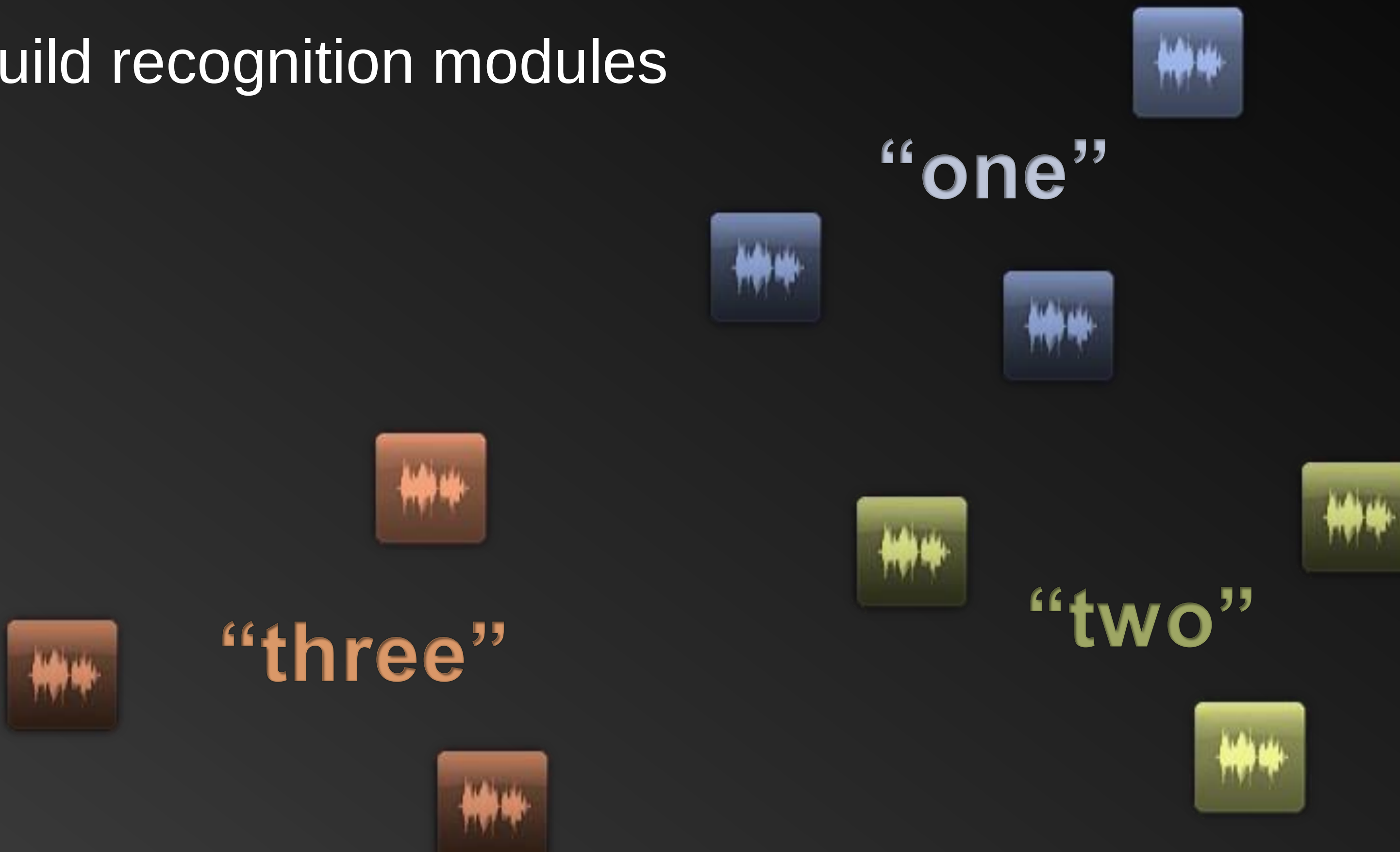


Putting it together

- Isolated word recognition
  - Vocabulary of size M
- Acquire data
  - Training set of size L
  - Test set

Putting it together

- Build recognition modules



Putting it together

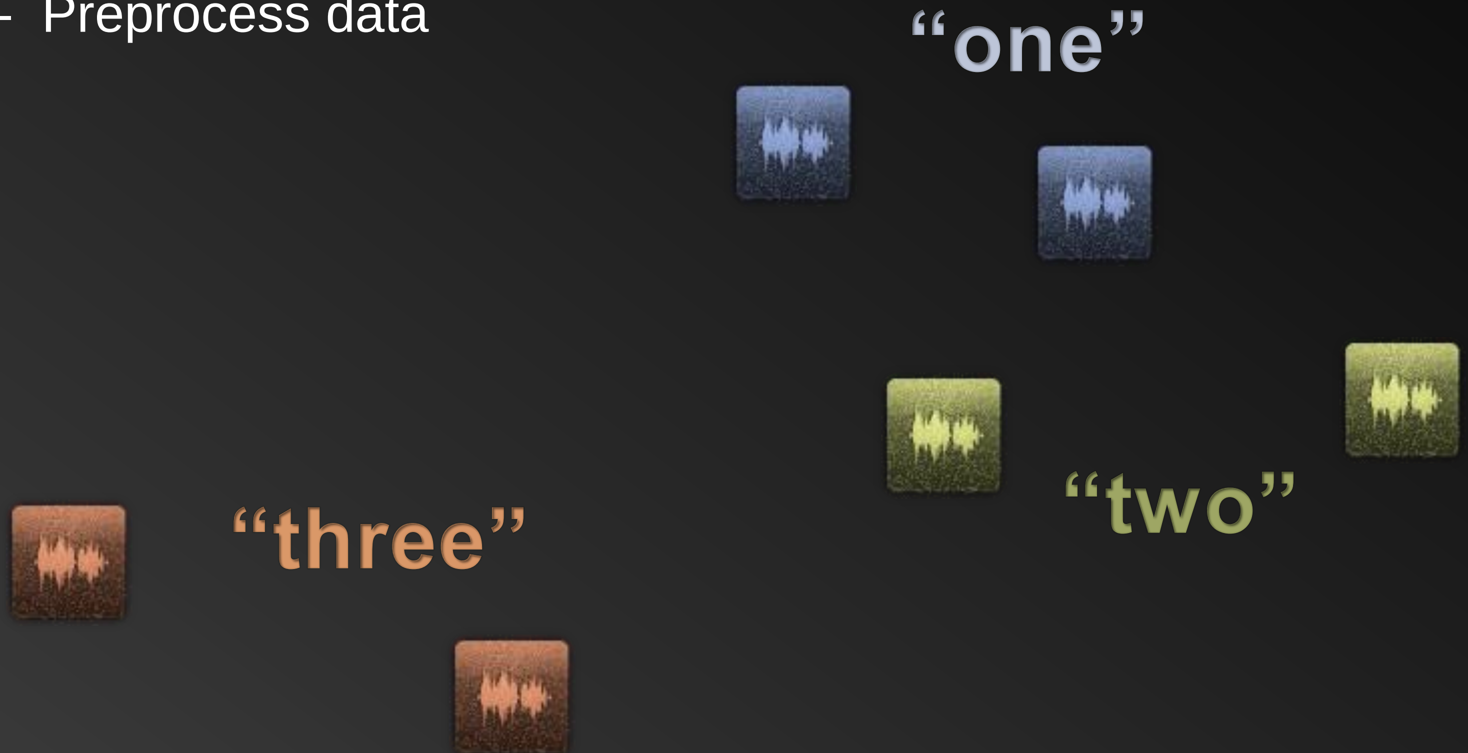
- Build recognition modules

Putting it together

- Build recognition modules

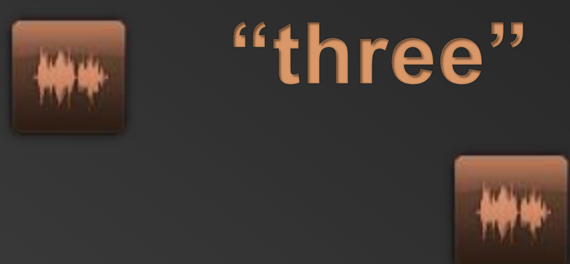
Putting it together

- Build recognition modules
  - Preprocess data



Putting it together

- Build recognition modules
  - Preprocess data
    - Filtering

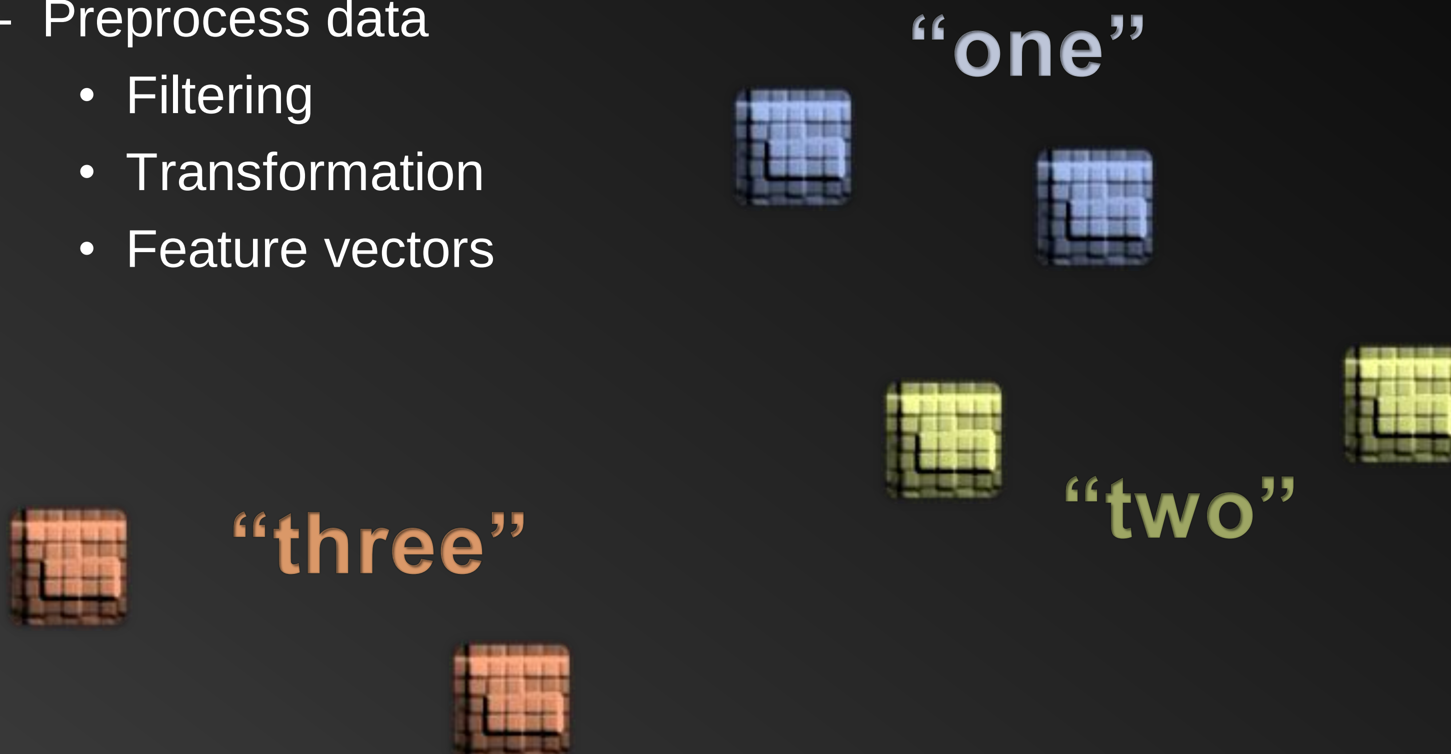


Putting it together

- Build recognition modules
  - Preprocess data
    - Filtering
    - Transformation

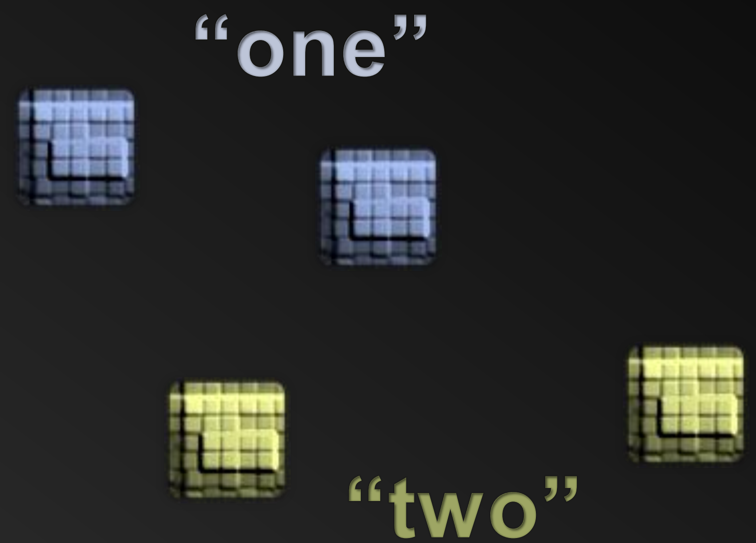
Putting it together

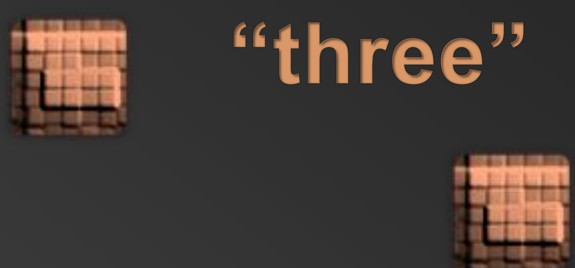
- Build recognition modules
  - Preprocess data
    - Filtering
    - Transformation
    - Feature vectors



Putting it together

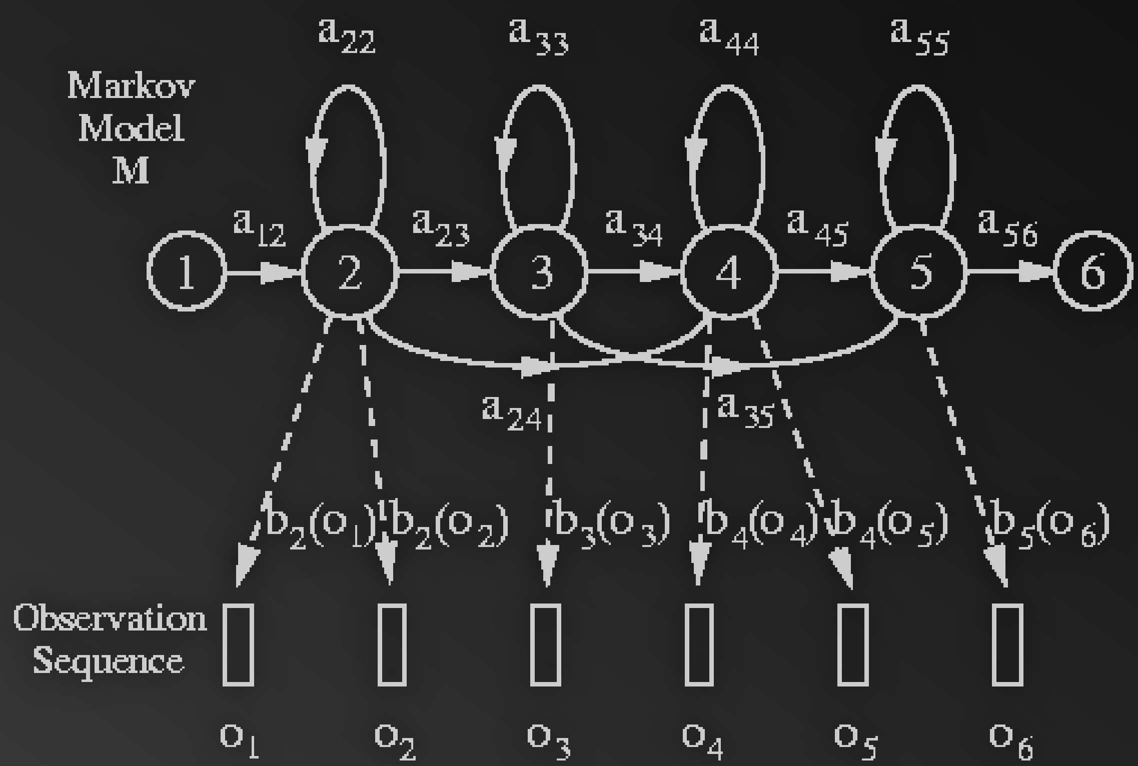
- Build recognition modules
  - Preprocess data
    - Filtering
    - Transformation
    - Feature vectors
  - Train HMM‘s





Putting it together

- Build recognition modules



Putting it together

- Build recognition modules
  - Preprocess data
    - Filtering
    - Transformation
    - Feature vectors
  - Train HMM‘s

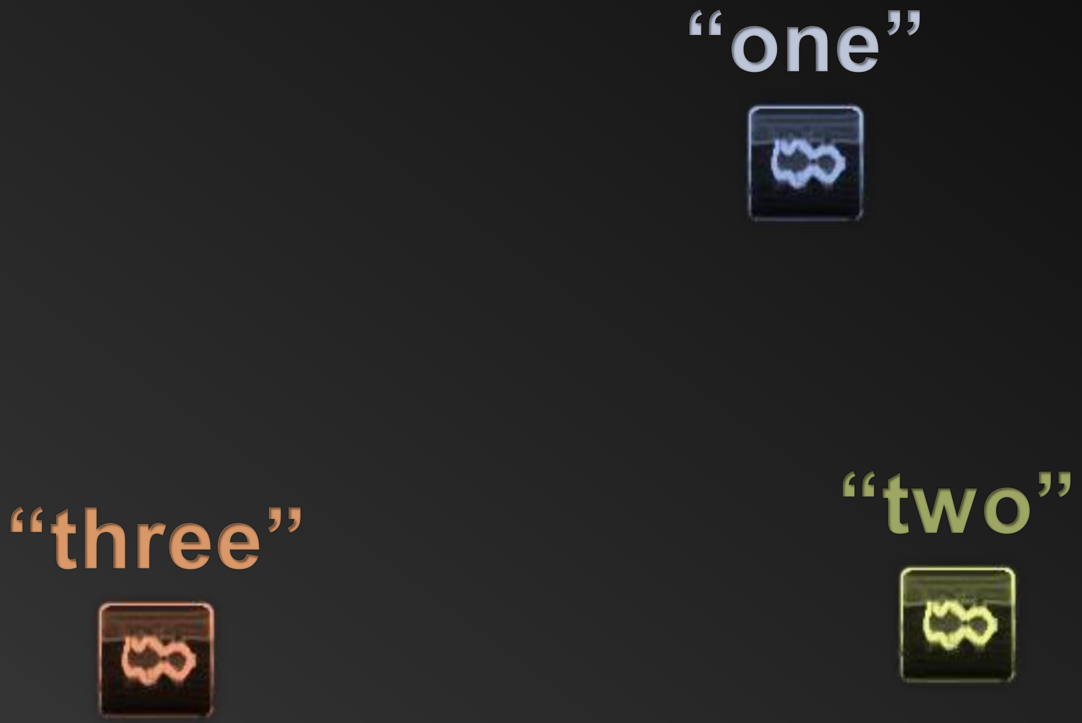
“one”

“three”

“two”

Putting it together

- Recognize



Putting it together

- Recognize

Putting it together

- Recognize

Putting it together

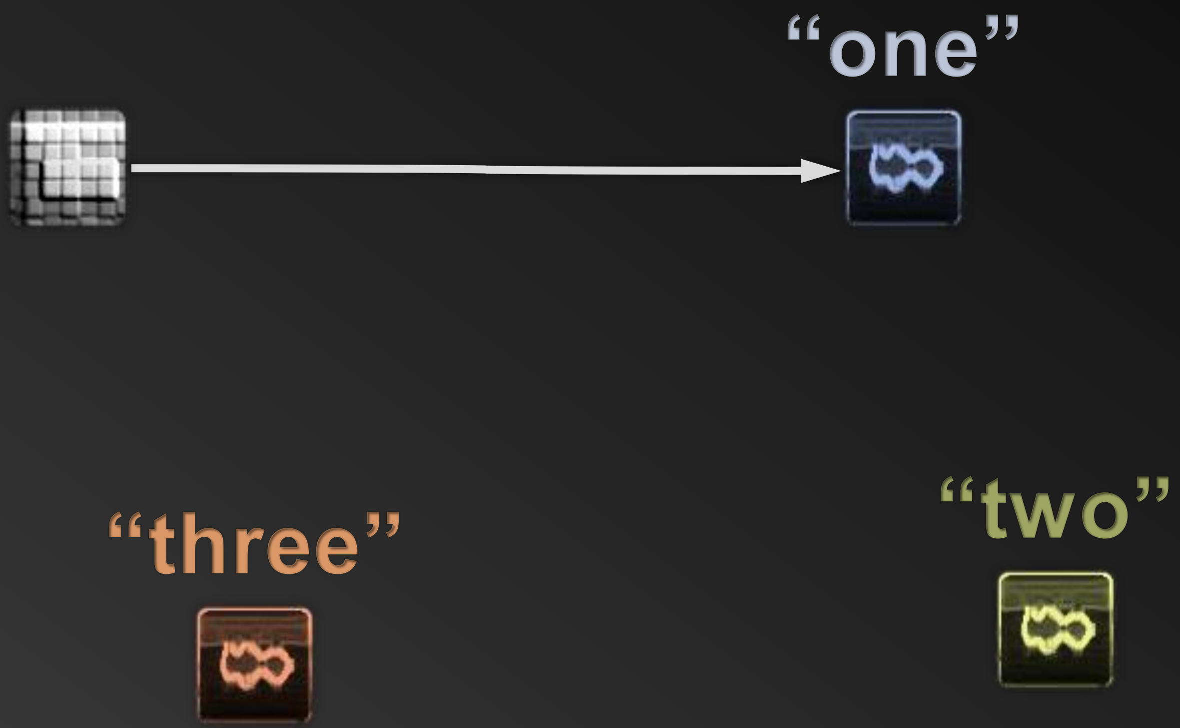
- Recognize

Putting it together

- Recognize

Putting it together

- Recognize



Putting it together

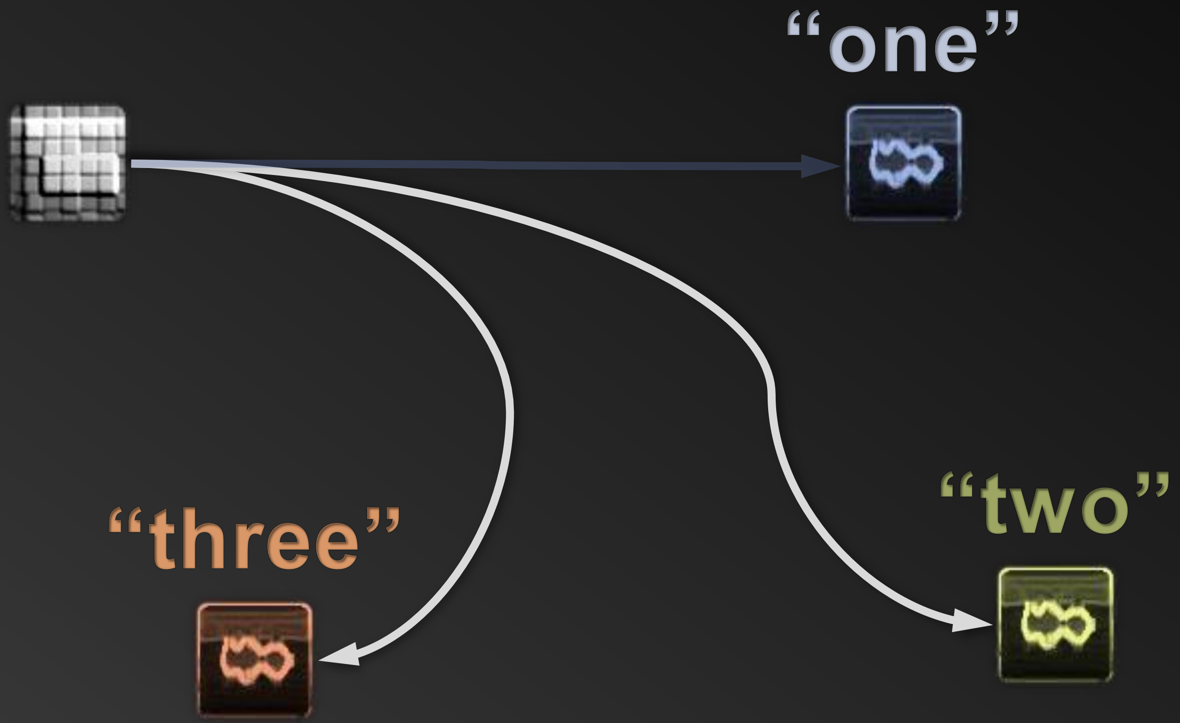
- Recognize

Putting it together

- Recognize

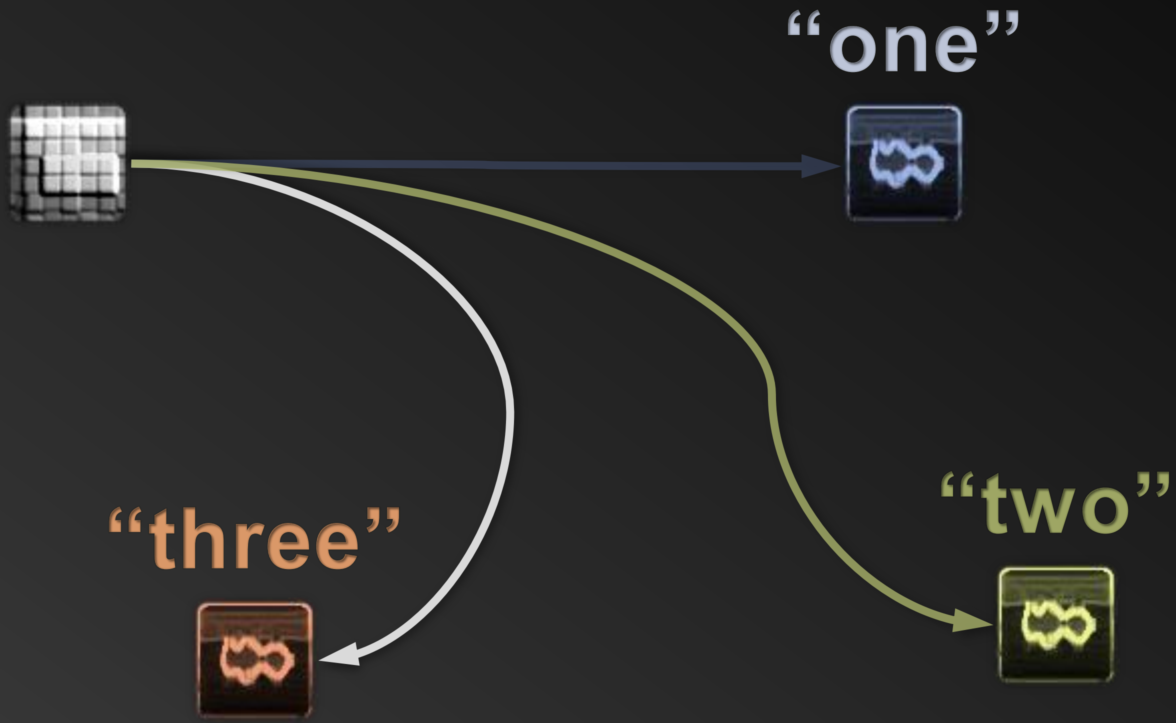
Putting it together

- Recognize



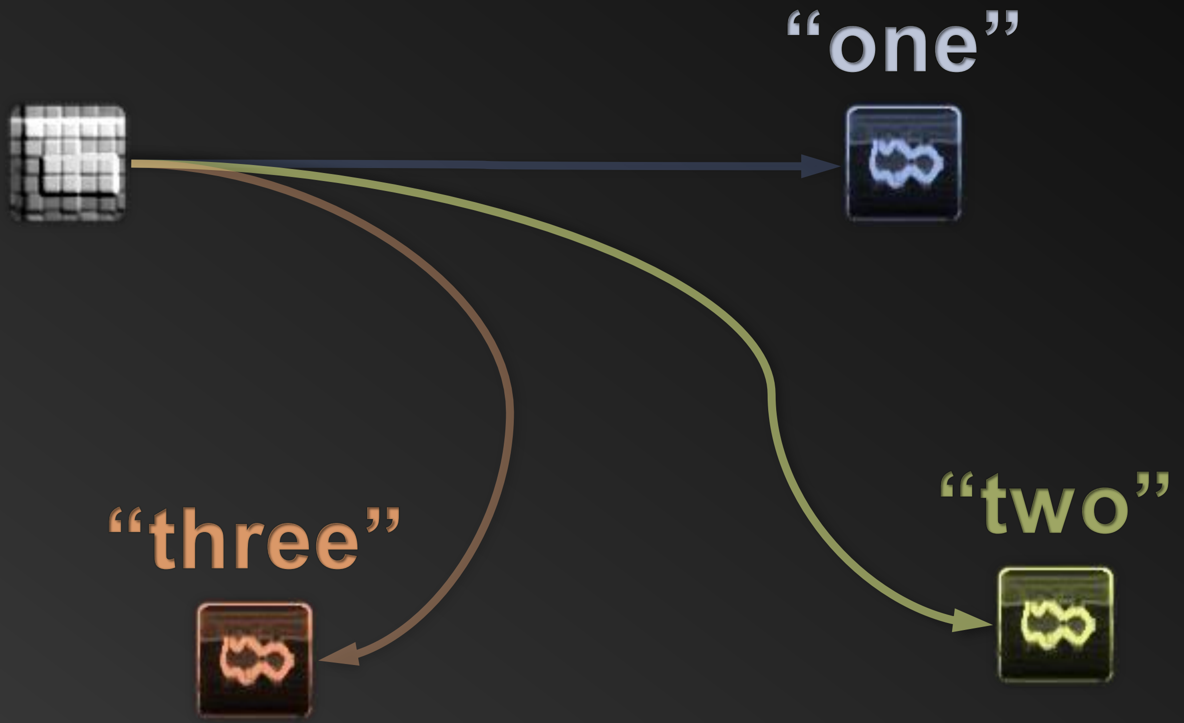
Putting it together

- Recognize



Putting it together

- Recognize



Putting it together

- Recognize

Putting it together

- Recognize

Putting it together

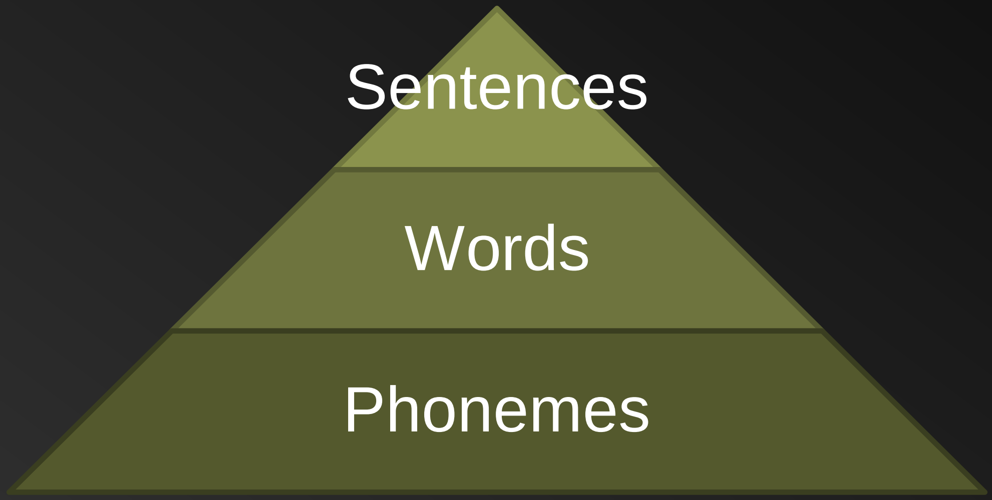
- Continuous speech recognition

Putting it together

- Continuous speech recognition
  - Hierarchical layers of specifically trained HMM‘s

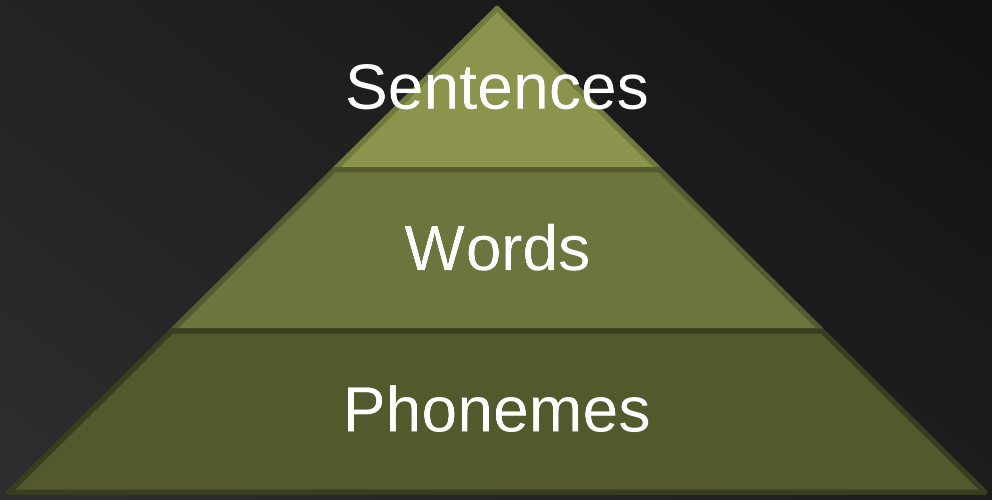
Putting it together

- Continuous speech recognition
  - Hierarchical layers of specifically trained HMM‘s



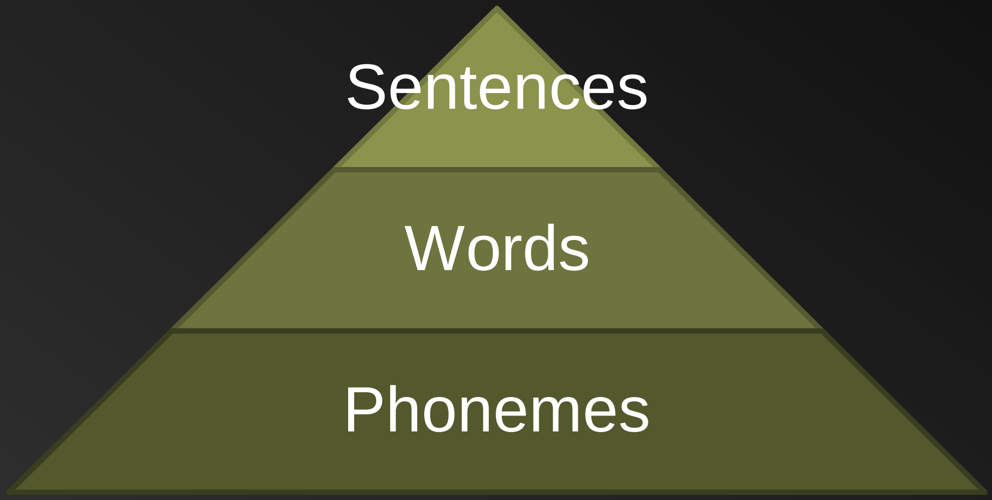
Putting it together

- Continuous speech recognition
  - Hierarchical layers of specifically trained HMM‘s
  - Extensive use of relational information



Putting it together

- Continuous speech recognition
  - Hierarchical layers of specifically trained HMM‘s
  - Extensive use of relational information
    - Diphones
    - Triphones
    - Grammar
    - N-Grams
    - NLP



Speech Recognition Today

Speech Recognition Today

- Research
  - DNNs for feature extraction and dimensionality reduction
  - Model improvement and automization
  - Audio processing (e.g. Hearbo)
  - Microsoft Translator
  - …

Speech Recognition Today

- Applications
  - Cars, Webbrowser, PC’s, Smartphones
  - iOS Siri
  - Dragon Naturally Speaking
  - Robot bartender JAMES
  - …

„Computer: end presentation.“